White Paper

From the Data Center to the Desktop: Putting High Performance Computing to Work

Table of Contents

A Practical Definition of HPC	**3**
The Payoff of HPC	**3**
Getting the Most from Compute Clusters	**3**
Choosing Your Nodes	**3**
Cost-Effective Clustering	**4**
Optimizing Applications	**4**
Compilers	4
Math Libraries	5
Debugging Tools	6
Transforming Research and Business Computing	**6**
Getting Started	**6**

In the past, high performance computing (HPC) required costly supercomputers that typically required whole buildings to house them, huge amounts of power and cooling to run, and whole teams of engineers to write programs for them. Today, new power-efficient, multi-core platform technology offers the possibility of using HPC in any environment, from clustered platforms in the enterprise data center to four or eight-processor systems on the researcher, analyst or designer's desktop.

The biggest challenge to creating these new, mainstream HPC systems is developing software that can take advantage of all the power of the new platforms. HPC requires parallel computing that can be complex and costly to develop, debug, deploy and maintain. The clusters typically used for HPC can be equally complex to architect and operate.

What will bring HPC to the mainstream are development tools that can automate the process of parallelizing applications, allowing developers to provide value through their domain expertise, and clustering solutions that streamline and simplify the process of building and maintaining compute clusters. Together, Dual-Core Intel® Xeon® processors and Microsoft® Windows® Compute Cluster Server 2003 (CCS) offer fast HPC development plus outstanding performance and reliability while accelerating time to insight with an HPC platform that is simple to deploy, operate, and integrate with existing infrastructure and tools.

A Practical Definition of HPC

HPC involves using a collection of computing resources that cooperate to solve a problem involving large amounts of data and calculation. The simplest and most cost-effective platform for HPC applications is to use a cluster of connected, independent multi-core and multi-threaded computers. The computing approach may be "tightly coupled," as in symmetric multi-processing, where computing elements do synchronized work on parts of the same task, or more "loosely coupled," like the computers in a cluster that work independently and synchronize only as required by the computing problem (for example, to compile the sum of independent calculations).

HPC generally demands specialized program optimizations to get the most from a system in terms of input/output, computation, and data movement. Some of these optimizations are handled by the commercial, scientific or other HPC applications, and some must be handled by the supercomputer operating system or the cluster server.

The Payoff of HPC

The payoff of HPC depends on the number and performance of available computing resources and the developer's ability to partition a computing problem or set of tasks to take advantage of the available resources. Amdahl's Law predicts the amount of speed-up possible by parallelizing any given problem. It basically says that the speed-up to be gained through parallelizing a problem is a function of the number of nodes available to share the work, their speed, and the percentage of the calculations that can be performed in parallel vs. the percentage that must be performed serially.

Parallelization can happen at many levels: OS, clustering, the application, etc. In most organizations, developers have acquired a great deal of expertise in their scientific, engineering, or business domain. Parallel programming and cluster management for HPC requires a different set of expertise. It can be time-consuming and expensive to build, test and optimize parallel applications from the ground up. Organizations will achieve the fastest and highest payoff from HPC by choosing platforms that simplify and automate the process as much as possible.

Getting the Most from Compute Clusters

Clustered platforms are the most cost-effective way to do HPC, especially if the clusters are built from affordable, industry-standard components that allow developers and system managers to leverage existing skills and infrastructure. There are three ways to get the most from HPC clusters:

- Choosing robust, high performance platforms for the nodes in your cluster
- Minimizing Total Cost of Ownership (TCO)—the cost to deploy, run, manage, and expand the HPC system—by choosing a clustering solution that is simple to set up and manage
- Maximizing utilization through robust, efficient clustering software and parallel programming

For example, the combination of Dual-Core Intel Xeon processors and CCS meets these goals with outstanding HPC performance and industry-leading reliability—clustering that is simple to deploy, operate, and integrate with existing infrastructure, and integrated parallel programming tools to accelerate the programming and deployment process.

Choosing Your Nodes

Processors are the foundation of your HPC system: Amdahl's Law tells us that the potential speed-up of your applications is determined in part by the performance of the compute nodes in the cluster. When you choose a clustering platform, look for products that offer:

- Balanced performance. Performance affects both speed-up and solution costs. In choosing a cluster platform for your HPC application, consider raw performance for your application, and also look at price/performance and performance/watt and how they will affect TCO. Choose the most affordable platform that will meet your current requirements and scale to meet your future HPC needs.
- Best-in-class reliability. System downtime adds to your time-to-solution, and computer worms, viruses or data errors can also affect the results of complex applications that may take hours or even days to re-run. To prevent these problems, choose cluster nodes that offer advanced reliability features such as enhanced memory ECC, sparing and mirroring for data protection and availability, and symmetric CPU access for fast system restart and/or processor failover.
- Widespread vendor relationships that give you choice and flexibility in system components and software
- HPC expertise and comprehensive support for developers creating HPC applications and systems

New 95W Dual-Core Intel Xeon processors offer up to twice the performance of previous processors while boosting power efficiency by up to 70%.[1] In addition, rack-optimized versions of Intel Xeon processors at 65 watts and low-voltage processors at 40 watts for ultra-dense deployments provide yet more options to optimize for power and performance efficiency. Advanced reliability features help provide the data integrity and stability that HPC applications require. 64-bit Intel® processors are supported by a wide ecosystem of hardware and software vendors, and Intel offers comprehensive developer support, including Intel® Software College, a Threading Immersion Program, and Application Tuning Centers worldwide.

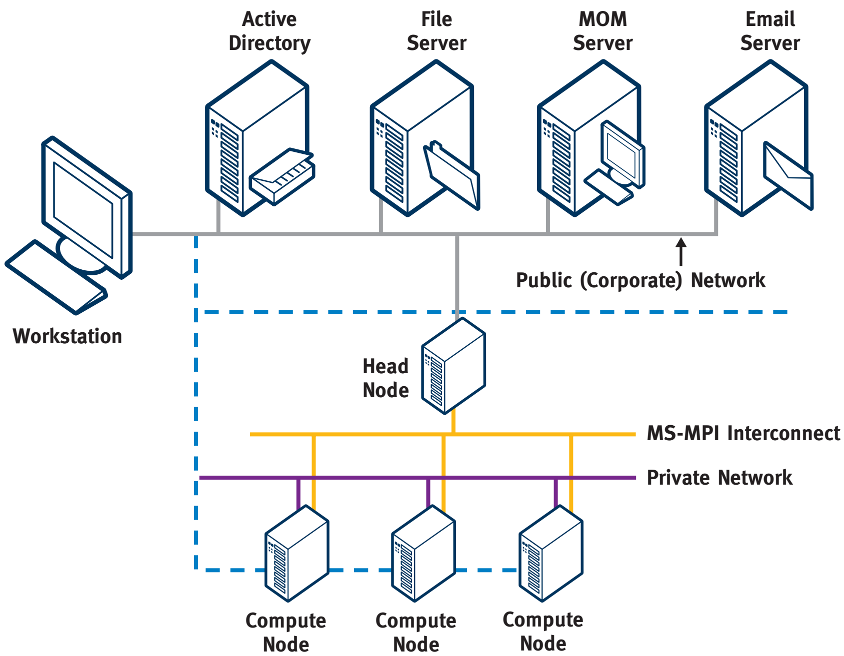

Figure 1: Architecture of Windows® Compute Cluster Server 2003

Cost-Effective Clustering

Clustering is a means to an end, so your clustering solution should simplify the process of deploying and managing your HPC applications. Your chosen clustering solution should offer:

- Ease of configuration and deployment
- Ease of management (reconfiguring or redistributing resources; problem identification, avoidance, and failover; ease of upgrades)
- Ease of expansion

CCS includes prescriptive setup procedures, a complete suite of management tools, and an integrated Job Scheduler. CCS is integrated with Active Directory® and uses Microsoft® Management Console (MMC) and Microsoft Operations Manager® (MOM) to provide a simple and familiar interface for managing and administering the cluster.

As shown in Figure 1, Windows CCS architecture simplifies cluster management, using a head node that provides the single point of management, deployment, and job scheduling for the compute cluster. This node controls and mediates all access to the cluster resources, using the existing corporate infrastructure and Active Directory for security and account management, and manages operations using tools such as MOM 2005 and System Management Server (SMS) 2003.

CCS integration with tools like Visual Studio® 2005 supports parallel job development and debugging and allows developers to use existing Windows-based skills and experience. CCS includes both a command-line job scheduler and the Compute Cluster Manager. The Command-Line Interface (CLI) supports Perl,* Fortran,* C/C++, C#, and Java.*

Optimizing Applications

Development tools can greatly simplify and speed the process of writing or converting applications to HPC. Ideally, development tools should be compatible with your current tools—IDEs, etc.—allowing developers to leverage their current skill sets. They should automatically perform optimizations that take advantage of hyper-threaded and multi-core architectures, and they should be integrated with your clustering solution. For example, Intel® Software Development products are optimized and integrated for CCS and fully integrated with the Visual Studio 2005 IDE, enabling developers to quickly migrate their applications to Windows-based HPC clusters.

Compilers

Intel and Microsoft compilers have a number of features that help produce optimized code for HPC-type applications. These optimizations range from memory hierarchy optimizations all the way to instruction scheduling at the pipeline level. Higher-level optimizations include, for example, loop transformations (unrolling, fusion, splitting, joining, etc). Lower-level optimizations address aspects of the particular chip being compiled for (e.g., register allocation, instruction scheduling, peephole optimizations, etc.). Floating-point (FP) optimizations are particularly important for HPC programs, and both sets of compilers do a good job of addressing the needs of FP-intensive code.

Programmer-assisted parallelization is supported by both sets of compilers through the OpenMP* standard. OpenMP is the industry standard for portable multithreaded application development. It is effective at fine-grain (loop-level) and large-grain (function-level) threading. OpenMP directives are an easy and powerful way to convert serial applications into parallel applications, enabling potentially big performance

gains from parallel execution on multi-core and symmetric multiprocessor systems. OpenMP enables developers to parallelize loops and code regions via *#pragma's* (compiler directives) in the source code. When compiled serially (e.g., at debug time), the pragmas are ignored and the code executes serially. This standard is supported by all major compilers and platforms.

Intel® compilers also support automatic parallelization. The Intel® C++ Compiler for Windows and Intel® Visual Fortran Compiler for Windows automatically optimize and parallelize software to take best advantage of multi-core Intel® processors. Through sophisticated dependence analysis, the compiler can figure out whether a loop can be parallelized safely without any #pragma directives. Automatic parallelization improves application performance on multiprocessor systems by means of automatic threading of loops. This option detects loops capable of being executed safely in parallel and automatically generates multithreaded code. Automatic parallelization relieves the user from having to deal with the low-level details of iteration partitioning, data sharing, thread scheduling, and synchronizations. It also provides the performance benefits available from multiprocessor systems and systems that support Hyper-Threading (HT) Technology. Intel C++ and Fortran compilers are integrated into the Microsoft Visual Studio 2005* development environment, the most widely used IDE in the industry, which provides a productive workplace for development, debugging, performance analysis and deployment of HPC programs. Visual Studio 2005 includes support for developing HPC applications, including parallel compiling and debugging, support for the development of applications for IA-32 and processors that support Intel® EM64T, as well as Itanium® 2 processors, and support for the industry-leading MPI2 message-passing standard. Integrated context-sensitive help is available from within Visual Studio.

Math Libraries

The use of pre-optimized math libraries can also speed HPC development and enhance performance, especially when the libraries are optimized for the underlying multi-core and/or hyper-threading platform. Intel® Math Kernel Library (Intel® MKL) is a set of highly optimized, thread-safe, mathematical functions for engineering and scientific applications that need maximum performance. Intel MKL supports Microsoft MPI, and the Cluster Edition of Intel MKL also includes ScaLAPACK and distributed memory Fast Fourier Transforms for Microsoft Windows clusters.

2-D Fast Fourier Transform

Single-Precision Complex-Data (In Place)

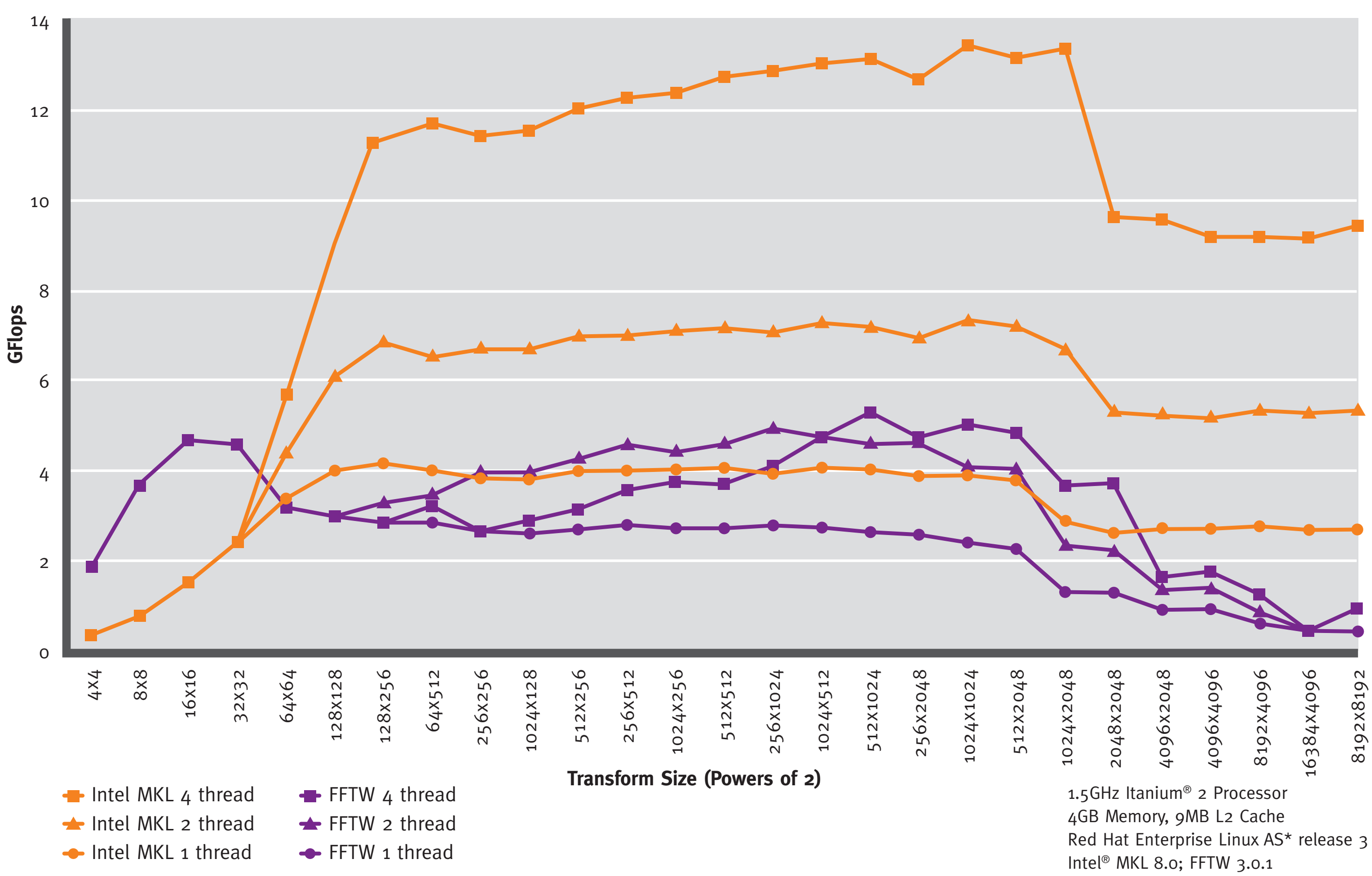

Figure 2: Intel® Math Kernel Library is highly optimized for clustered, multi-core and hyper-threaded platforms. This optimization helps to significantly accelerate HPC functions such as Fast Fourier Transforms.

Intel MKL is threaded to make maximum use of system and platform processors like Intel's new Intel® Core™ Duo processors. Here are some of the PC resource and performance issues that Intel MKL optimizes for:

- CPU—register use, FP units
- Cache—keep data in cache as long as possible; deal with cache interleaving
- Translation Look-ahead Buffer—maximally use data on each page
- Memory bandwidth—minimally access memory
- Computer—use all available processors via threading
- System—use all the nodes available

Intel® MKL contains powerful math functions including:

- Linear Algebra—BLAS & LAPACK & ScaLAPACK
- Linear Algebra—Sparse Solvers (direct and iterative)
- Fast Fourier Transforms—shared memory and clusters
- Vector Math
- Vector Random Number Generators

Debugging Tools

Debugging parallel applications can be challenging, especially when threads can be spread across multiple nodes in a cluster, multiple CPUs in each node, and multiple threads within each CPU. Developers need a debugger that gives clear visibility and non-invasive control over executing threads.

The Microsoft Visual Studio IDE provides a Parallel Debugger that can be used effectively in the Windows CCS environment. The debugger allows monitoring of multiple threads running on different machines via remote agents in one window, while providing access to all the standard features of the Visual Studio debugger. While the Parallel Debugger does a good job of addressing typical-size clusters, debugging across thousands of nodes is a challenge for any debugger on the market today. The Visual Studio team is actively researching this topic and enhancing its IDE to address developer needs in this critical area.

Transforming Research and Business Computing

High performance computing applications can offer invaluable insights into research, design and business questions. While HPC has traditionally been the province of large research organizations and corporations, the availability of mainstream HPC solutions opens a vast range of new possibilities:

- Even mid-sized businesses can afford real-time financial analysis and business intelligence applications that can project results of different product mixes and strategies.
- Businesses can do richer data mining to identify and respond to customer preferences and market trends, potential security threats, etc.
- HPC creates new opportunities for services using expert systems in industries such as healthcare, entertainment, retailing and travel.
- Scientific and engineering organizations, colleges and universities can empower designers and engineers with more power to do simulation and analysis in aerospace, biosciences and pharmaceuticals, and other fields.

Together, CCS and Dual-Core Intel Xeon processor-based server systems offer simple and more affordable HPC solutions to help make mainstream HPC applications possible.

Getting Started

Your system vendor or business software vendor can help you get started with HPC solutions. Microsoft and Intel®-based solutions include advanced system management features that make them affordable to own and operate, and with industry leaders Microsoft and Intel, you can choose solutions from a vast ecosystem of vendors worldwide. Intel® Solution Services is Intel Corporation's worldwide professional services organization that helps companies capitalize on the full value of Intel® architecture through consulting focused on architecture transitions. Through Intel Solution Services, your development team gains expertise in Intel architecture and next-generation technologies to design cost-effective solutions that help deliver superior business results.

Microsoft Services is the consulting, technical support, and customer service arm of the world's leading software company. The organization helps customers and partners discover and implement high value Microsoft solutions that generate rapid, meaningful, and measurable results. With its global partner network and support infrastructure, Microsoft Services enables the successful adoption, deployment, and use of Microsoft solutions and technologies for all customers, from the individual to the enterprise.

For more information about Windows Compute Cluster Server 2003, visit www.microsoft.com/hpc. For information about purchasing Compute Cluster Server, email us at hpcinfo@microsoft.com or to find Microsoft offices worldwide, visit www.microsoft.com/worldwide.

Accelerate your transition to Intel dual-core processing. Talk with your local Intel representative, and visit us on the web at www.intel.com/go/xeon. And for more information about Intel® platforms in high performance computing, go to www.intel.com/go/hpc.

Optimize application performance on HPC systems and multi-core processors with Intel Software Development products. Learn more at www.intel.com/software/products.

For additional information on Intel Solution Services, download a brochure or contact us at www.intel.com/go/intelsolutionservices.

For information about Microsoft Services, visit www.microsoft.com/services/microsoftservices/srv_about.mspx.

[1] Performance tests and ratings are measured using specific computer systems and/or components and reflect the approximate performance of Intel products as measured by those tests. Any difference in system hardware or software design or configuration may affect actual performance. Buyers should consult other sources of information to evaluate the performance of systems or components they are considering purchasing. For more information on performance tests and on the performance of Intel products, visit http://www.intel.com/performance/resources/limits.htm or call (U.S.) 1-800-628-8686 or 1-916-356-3104.

Printed in USA/0606/MH/PMS/PDF   Order Number: 313837-001US

www.microsoft.com

Microsoft Corporation
One Microsoft Way
Redmond, WA 98052-6399
1-800-MICROSOFT (1-800-642-7676)

www.intel.com

Intel Corporation
2200 Mission College Blvd.
Santa Clara, CA 95052